A Framework for Multi-objective SLA Compliance Monitoring

Joel Sommers*, Paul Barford*, Nick Duffield†, Amos Ron*
*University of Wisconsin-Madison
†AT&T Labs–Research

***Abstract*—**
Service level agreements (SLAs) specify performance guarantees made by service providers, typically in terms of packet loss, delay, delay variation, and network availability. While many tools have been developed to measure individual aspects of network performance, there has been little work to directly address the issue of SLA compliance monitoring in an operational setting where accuracy, parsimony, and other related issues are of vital importance. This paper takes the following steps toward addressing this problem: (1) we introduce an architectural framework for integrating multiple discrete-time active measurement algorithms, an architecture that we call *multi-objective monitoring*; and (2), we introduce a new active measurement methodology to monitor the packet loss rate along a network path for determining compliance with specified performance targets which significantly improves accuracy over existing techniques. We present a prototype implementation of our monitoring framework, and demonstrate how a unified probe stream can consume lower overall bandwidth than if individual streams are used to measure different path properties. We demonstrate the accuracy and convergence properties of the loss rate monitoring capability of SLAM in a controlled laboratory environment using a range of background traffic scenarios and examine its accuracy improvements over existing techniques.

I. Introduction

IP networks have become indispensible to businesses, governments, and individuals, worldwide. Reflecting this importance, it is increasingly common for service providers to offer transport-level performance guarantees using metrics such as packet loss, delay, and network network availability as part of their service level agreements (SLAs) [1]–[4]. Meeting agreed-upon performance targets results in the collection of revenue for the service provider, whereas not meeting these objectives can result in credits and loss of revenue to the customer. Accurate network monitoring for the purpose of detecting compliance with performance goals is therefore critical to both parties.

The problem of monitoring compliance with agreed-upon performance metrics is a key challenge of SLA engineering. A provider must design SLAs that can be accurately and efficiently monitored, while at the same time minimizing the possibility of non-compliance. For example, guaranteeing a very low loss rate might be possible only if loss rates can be estimated in a lightweight way with sufficiently high confidence. While passive measurements (*e.g.*, SNMP MIB counters) may provide adequate accuracy for a metric such as loss on a link-by-link basis, they are insufficient for estimating the actual performance experienced by customer traffic (*e.g.*, due to dynamic routing changes or hardware failures). Thus, although there are situations where active measurements may be too heavyweight or may yield inaccurate results [5]–[7], they nonetheless remain a key mechanism for SLA compliance monitoring.

In this paper, we address the following question: can SLA compliance be accurately monitored with a single lightweight probe stream? There have been a number of active measurement tools and methodologies proposed over the years to estimate transport-level performance characteristics. Even so, there has been little work to directly address the problem of SLA compliance monitoring. In this context, measurement tool accuracy, parsimony, ability to report confidence bounds, and ability to quickly adapt to changing network conditions are of great importance.

The first contribution of this work is the introduction of a framework for integrating multiple discrete time-based active measurement algorithms. Modules for estimating individual path characteristics interact with a central probe scheduler such that a given probe may be used for multiple purposes. The result is a unified probe stream that can consume lower overall bandwidth than if individual streams are used. Moreover, each module operates independently, thus preserving desirable statistical and accuracy properties for each estimation method. We describe the implementation of our framework in a tool called SLAM (SLA Monitor).

The second contribution of this paper is the introduction of a new active measurement methodology for estimating end-to-end packet loss rate. Starting from the geometric probe methodology described in [7], we develop a heuristic technique for estimating packet loss rate along a path that significantly improves accuracy over existing approaches. We implement this new methodology as a SLAM module.

We demonstrate the properties of SLAM in a controlled laboratory environment using a range of background traffic scenarios. We compare SLAM's loss estimation accuracy with both Poisson and periodic streams of the same rate, and examine the convergence and robustness of SLAM loss estimates. Our experiments reveal that SLAM estimates the end-to-end loss rate with high accuracy and with good confidence bounds. For example, in a scenario using self-similar background traffic, the true loss rate over a 15 minute period is 0.08% and the SLAM estimate is 0.07%. In contrast, Poisson and periodic methods for estimating loss rate have errors of more than two orders of magnitude.

II. Related Work

While many details of SLAs are considered proprietary, general aspects and structure of SLAs are discussed in [1], [8]. Performance guarantees associated with SLAs range from network path availability, to transport-related metrics such as packet loss, to application-specific metrics such as web response times and voice stream quality. Such guarantees may be based on various statistics of the given metric, such as the mean, median, or a high quantile such as the 95th percentile, computed over various time scales. Examples of the types of performance assurances offered by commercial providers are available online [2]–[4].

To ensure that SLA performance targets are met with high probability, service providers collect measurements either passively within the network, by injecting measurement probes into the network, or by using a combination of both [9]–[12]. While active measurement-based compliance monitoring has received some attention in the past, *e.g.*, [9], there has been little validation in realistic environments where a reliable basis for comparison can be established. Furthermore, practical issues such as balancing the impact of measurement tools on the network with estimation accuracy have seen less attention. Our work also takes an active measurement approach, introducing a framework for simultaneous, or multi-objective, measurement of transport-level performance metrics which can reduce the overall impact of the measurement process. We further differentiate our work through validation in a controlled, realistic testbed.

There has been a great deal of work on the problem of measuring end-to-end packet loss, *e.g.*, [13]–[20]. While there has been limited work addressing the accuracy of common measurement approaches, exceptions are found in [5]–[7]. The issue of accuracy clearly has serious implications for SLA compliance monitoring.

III. Multi-objective Probing

In this section, we introduce an architectural framework for integrating multiple discrete-time active measurement algorithms in a single probe scheduler to provide *simultaneous* estimation different network path properties.

Consider an ISP that wishes to monitor packet loss using the algorithm of [7], and simultaneously monitor packet delay and delay variation. Assume that the packet delay and delay variation algorithms operate in discrete time. A typical approach is to use three separate probe streams for monitoring these properties. However, since these algorithms operate in discrete time we may take advantage of the fact that they may send probes at the same time slot. We can accommodate such requests by tagging probes according to the estimator to which they apply. The effect is that a single probe packet may be used for multiple estimation objectives, thereby reducing overall impact of measurement traffic on the network. This is the intuition behind multi-objective probing.

The basic architecture of our multi-objective probe scheduler is depicted in Figure 1. The central component of the architecture is a scheduler operating in discrete time that provides callback and probe scheduling mechanisms. Independent probe modules interact with the scheduler to implement particular estimation algorithms, *e.g.*, BADABING [7] Our probe scheduler design allows for logical separation among multiple, simultaneously operating measurement methods and for optimizations of network bandwidth. We implemented this architecture in a tool called SLAM (SLA MONITOR). SLAM sends UDP packets in a one-way manner between a sender and receiver. The scheduler consists of about 1,500 lines of C++.

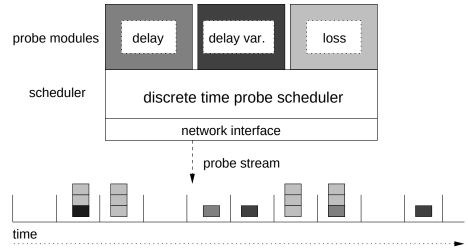

Fig. 1. Multi-objective probe scheduler architecture. Algorithmic modules interact with a generic discrete-time probe scheduler to perform estimation of delay, delay variation, loss characteristics, or other properties of interest.

Two important implementation decisions were made in the SLAM probe sender. First, the scheduler must accommodate estimation techniques that use multi-packet probes, such as BADABING [7] which uses them to obtain an improved estimate of instantaneous congestion. Second, the scheduler must arbitrate among probe modules that may use different packet sizes. At present, the smallest packet size scheduled to be sent at a given time slot is used.

For example, suppose three packets of size 600 bytes have been scheduled to be sent at time slot i for loss estimation and that one packet of size 100 bytes has also been scheduled for the same time slot i for delay estimation. When time slot i arrives, the scheduler will send a sequence of three packets of sizes 100, 600, and 600 bytes. The first packet will tagged for delay estimation, and all three packets will be tagged for loss estimation. At the receiver (assuming these packets are not lost in transit), the delay estimator module will receive one packet of size 100 bytes, and the loss estimator module will receive three packets of sizes 100, 600, and 600 bytes. We discuss implications of these implementation decisions in Section V.

IV. Packet Loss Rate Monitoring Methodology

We now describe the basic assumptions and method for estimating packet loss rate along an end-to-end path. Our objective is to develop an accurate, robust estimator based on a discrete-time probe process to be implemented as a module of SLAM.

The methodology described in [7] was shown to yield accurate estimates of congestion event frequency ($\widehat{F}$) and duration ($\widehat{D}$) along an end-to-end path. It was noted that the primary difficulty in estimating end-to-end *packet loss rate* (number of lost packets divided by total number of packets over a given time interval)—the loss performance metric specified in SLAs—is that it is unclear how to measure *demand* along the path, particularly during congestion periods. Therefore, we propose the following heuristic approach.

Starting from the geometric probe stream in [7], which initiates a probe pair at a given time slot with probability p_{loss}, we measure the loss rate $\widehat{l}$ of the probes *during congestion episodes*. Since the estimation techniques in [7] do not directly identify individual congestion episodes we take an empirical approach, treating consecutive probes in which at least one packet is lost as indication of a congestion episode. As in [7], we assume that the end-to-end loss rate L is stationary and ergodic. Given an estimate of the frequency of congestion $\widehat{F}$, we estimate the end-to-end loss rate as

$$\widehat{L} = \widehat{F}\widehat{l}.$$

The key assumption of this heuristic is that we treat the probe stream as a *marker flow*, namely, that the loss rate observed by this flow has a meaningful relationship to other flows along the path. As a basis for this assumption, we note that the probes in [7] consist of multiple packets (3 by default), which has some similarity to a TCP stream where delayed ACKs cause a sender to release two very closely-spaced packets. While we do not claim that the probe stream is, in general, the same as a TCP stream, our results below demonstrate that such an assumption may be reasonable in this context.

Finally, we note that using previous work which analyzed the variance of the frequency estimator, we can similarly derive confidence intervals on this loss rate estimator (details omitted due to space constraints) [21].

V. SLAM EVALUATION

We now describe the experimental evaluation of SLAM in a controlled laboratory environment. In our experiments, we fixed the SLAM loss rate module with parameter $p_{loss} = 0.3$ and packet sizes of 600 bytes, unless otherwise specified. These settings were found to give good loss characteristic estimates [7]. We verified the results regarding the setting of the parameter p_{loss} but omit detailed results in this paper.

A. Testbed and Traffic Scenarios

Our laboratory testbed, depicted in Figure 2, consisted of commodity workstation end hosts and commercial IP routing systems configured in a dumbbell-like topology. We used 10 workstations on each side of the topology for producing background traffic and one workstation at each side to run SLAM. Each workstation has a Pentium 4 processor running at 2GHz or better, with at least 1 GB RAM and an Intel Pro/1000 network interface card and was configured to run either FreeBSD 5.4 or Linux 2.6. The SLAM hosts were configured with a default installation of FreeBSD 5.4. Background traffic and probe traffic flowed over separate paths through a Cisco 6500 enterprise router (hop A) and was multiplexed onto a bottleneck OC3 (155 Mb/s) link at a Cisco GSR 12000 (hop B). Packets exited the OC3 via another Cisco GSR 12000

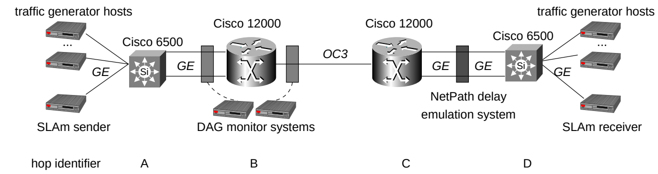

Fig. 2. Laboratory testbed. Probes and cross traffic are multiplexed onto a bottleneck OC3 (155Mb/s) link. Synchronized Endace DAG monitors are used to collect traces for calculation of true loss and delay values.

(hop C) and passed to receiving hosts via a Cisco 6500 (hop D). NetPath [22] is used between hops C and D to emulate propagation delays for the background traffic hosts in the testbed. We used a uniform distribution of delays with a mean of 50 msec, minimum of 20 msec, and maximum of 80 msec. The bottleneck output queue at the Cisco GSR at hop B was configured to perform tail drop with a maximum of 624 packets of size 1500 bytes, or about 50 msec of buffer space at 155 Mb/s. The SLAM workstations were synchronized to a Stratum 0 NTP server configured with a TrueTime GPS card. We used the synchronization software developed by Corell *et al.* [23] to provide accurate timestamps for SLAM. All management traffic for the systems in Figure 2 flowed over a separate network (not pictured in the figure).

An important aspect of our testbed is the ability to establish a reliable "ground truth" for our experiments. Optical splitters were attached to the links between hops A and B and to the link between hops B and C and synchronized Endace DAG 4.3 (Gigabit Ethernet) and 3.8 (OC3) passive monitoring cards were used to capture packet traces entering and leaving the bottleneck node. By comparing packet header information, we were able to identify which packets were lost at the congested output queue during experiments.

We used four background traffic scenarios in our experiments. For the first scenario, we used Iperf [24] to produce constant-bit rate (CBR) UDP traffic for creating a series of approximately constant duration (about 65 msec) loss episodes that were spaced randomly at exponential intervals with mean of 10 seconds over a 10 minute period. The second scenario consisted of 100 long-lived TCP sources run over a 10 minute period. For the final two scenarios, we used Harpoon [25] with a heavy-tailed file size distribution to create self-similar traffic approximating a mix of web-like and peer-to-peer traffic commonly seen in today's networks. We used two different offered loads of 60% and 75% of the bottleneck OC3. Experiments using the self-similar traffic scenario were run for 15 minutes. For all scenarios, we discarded the first 30 and last 30 seconds of the traces. Note that the SLAM parameters used in our experiments result in only about 0.3% of the bottleneck OC3 consumed for measurement traffic.

B. Multi-Objective Probing Evaluation

We first evaluate the bandwidth savings that can arise due to multi-objective probing. As we noted in Section III, if multiple probe modules each wish to send a probe at a given time slot, the smallest packet size of each of the modules is used.

An effect of this implementation decision is that the overall bandwidth requirement for the multi-objective stream may be less than the aggregate bandwidth requirement for individual probe modules, were they to be used separately.

Assume that we wish monitor packet loss rate using the algorithm described in Section IV. Assume also that we wish to send a fixed-rate periodic probe stream for monitoring, *e.g.*, delay or delay variation. We set the probe packet sizes at 600 bytes for the loss probe and 100 bytes for the periodic probe. We compare probe rates using two different parameter sets: in parameter set A, p_{loss} is 0.3 and the periodic probe interval is 100 milliseconds, and for parameter set B, p_{loss} is 0.2 and the periodic probe interval is 20 milliseconds. Table I shows the results for these experiments. The table shows, for example, that for parameter set A, the the loss probe stream is separately about 345 Kb/s, and the delay probe stream is about 40 Kb/s: a sum of 385 Kb/s. With SLAM, the probe stream is about 297 Kb/s, a savings of 23%. While the savings is parameter dependent (as shown in the table), there are clearly obtainable bandwidth savings.

TABLE I

EXAMPLES OF AVERAGE BANDWIDTH REQUIREMENTS FOR INDIVIDUAL MEASUREMENT METHODS AND FOR MULTI-OBJECTIVE PROBE STREAM. THE DISCRETIZATION TIME INTERVAL IS SET TO 5 MILLISECONDS, AND PROBE PACKET SIZES ARE CHOSEN TO BE 600 BYTES FOR THE LOSS PROBE AND 100 BYTES FOR A PERIODIC PROBE STREAM. FOR PARAMETER SET A, p_{loss} IS SET TO 0.2 AND THE PERIODIC PROBE INTERVAL IS SET TO 20 MILLISECONDS. FOR PARAMETER SET B, p_{loss} IS SET TO 0.3 AND THE PERIODIC PROBE INTERVAL IS SET TO 100 MILLISECONDS. ALL VALUES ARE IN KB/S.

Parameter set	Loss	Periodic stream	Sum (separate streams)	SLAM	Savings
A	345	40	385	297	88 (23%)
B	489	8	497	474	23 (5%)

C. *Loss Rate Estimation Accuracy*

We now examine the accuracy of the loss rate estimates for SLAM, comparing SLAM's accuracy with standard Poisson-modulated [20] and periodic streams of the same rate as the SLAM stream.

Table II compares the true loss rate measured using the passive traces with the loss rate estimates of SLAM and the Poisson and periodic probe streams. Values are shown for each of the four traffic scenarios and are average loss rates over the duration of each experiment. Note that differences in true values are due to inherent variability in traffic sources. We see that for all four scenarios, the Poisson and periodic streams yield very poor estimates of the true loss rate. In all but one case, the estimates are off by more than two orders of magnitude—a significant relative error. In fact, the Poisson and periodic estimates are generally close to zero—a phenomenon consistent with earlier experiments [7] and primarily due to the fact that single packet probes generally yield poor indications of congestion along a path. (Note that these accuracy improvements are consistent with experiments described in [7].) The estimates produced by SLAM are significantly better, with a maximum relative error in the case of the CBR background traffic. Both SLAM loss rate estimates for the self-similar background traffic have relative errors of about 10% or less.

TABLE II

COMPARISON OF LOSS RATE ESTIMATION ACCURACY FOR SLAM, POISSON, AND PERIODIC PROBE STREAMS. VALUES ARE AVERAGE LOSS RATES OVER THE FULL EXPERIMENT DURATION.

Probe stream →	SLAM		Poisson		periodic	
Traffic scenario ↓	true	estimate	true	estimate	true	estimate
CBR	0.0051	0.0073	0.0051	0.0017	0.0051	0.0017
Long-lived TCP	0.0163	0.0189	0.0163	0.0062	0.0163	0.0050
Harpoon self-similar (60% load)	0.0008	0.0007	0.0017	0.0000	0.0018	0.0000
Harpoon self-similar (75% load)	0.0049	0.0050	0.0055	0.0000	0.0060	0.0011

D. *Robustness of Loss Estimation*

Estimation accuracy over relatively long time periods (*e.g.*, 10 minutes) is clearly desirable from the standpoint of SLA compliance monitoring. Also important are the dynamic properties of an active measurement estimator, *i.e.*, how well the method adapts to changing network conditions and how quickly the estimator converges to the average path state. In this section, we examine the time varying nature of the SLAM estimates for packet loss.

Figure 3 shows the true loss rate and SLAM-estimated loss rate over the duration of experiments using long-lived TCP traffic (top) and self-similar traffic at 60% offered load (bottom). As above, true loss rate estimates are shown for 10 second intervals and estimates for SLAM are shown for 30 second intervals. Results for CBR traffic are not shown but are consistent with plots in Figure 3. The upper and lower bars for SLAM indicate estimates of one standard deviation above and below the mean using the variance estimates derived from [21]. For the SLAM estimates we see the narrowing of variance bounds as an experiment progresses, and that the true loss rate is, with few exceptions, within these bounds. We also see that SLAM tracks the loss rate over time quite well, with its estimated mean closely following the true loss mean.

VI. DISCUSSION AND CONCLUSIONS

SLA monitoring is of significant interest to both customers and providers to ensure that the network is operating within acceptable bounds. This paper introduces a new framework for multi-objective SLA compliance monitoring using active measurements and introduces a new method for measuring end-to-end packet loss rate. We implemented the probing framework and loss rate methodology in a tool called SLAM and evaluated the tool in a controlled laboratory setting. Our results demonstrate the bandwidth savings that can result due to multi-objective probing. Our results also show that SLAM packet loss rate estimates are much more accurate than loss rate estimates obtained through standard periodic or Poisson probe streams, and that these standard techniques may not

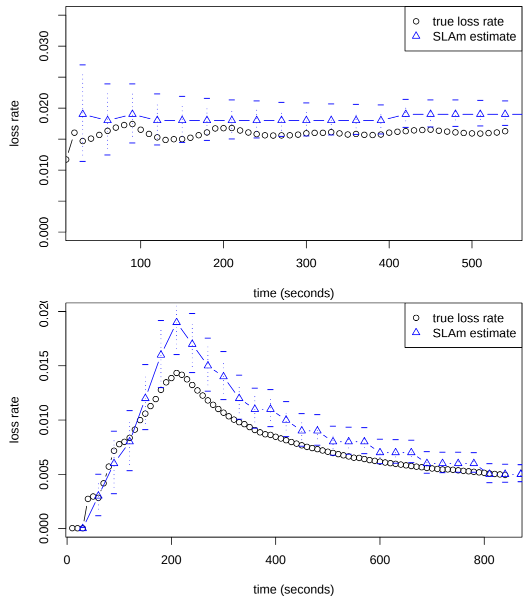

Fig. 3. Comparison of true loss rate with SLAM estimates over time. True loss rates are plotted using 10 second intervals. SLAM estimates are plotted using 30 second intervals. Plots shown for long-lived TCP (top) and self-similar traffic at 60% offered load (bottom) traffic scenarios. The upper and lower bars for SLAM indicate estimates of one standard deviation above and below the mean using the variance formulation of [21].

provide an accurate estimate of the state of the network, thereby preventing an accurate assessment of SLA compliance. Furthermore, we illustrated the convergence and robustness properties of the loss rate estimates of SLAM which make it useful in an operational setting.

We believe that SLAM represents a significant step toward accurate, low-impact SLA compliance monitoring using active measurements. However, there are a number of issues that this work does not address. First, there are several other end-to-end properties of interest for SLA compliance monitoring such as delay and delay variation. We intend to enhance SLAM to estimate these characteristics in the future. Second, our focus is on monitoring in the context of a single end-to-end path. In a typical operational settings, however, a network consisting of many links and paths must be monitored. In this context, a deployment strategy must be developed to coordinate probe streams so that links internal to the network are not carrying "too much" measurement traffic. A detailed analysis of this issue is a focus of future work. Next, our validation and calibration of SLAM is performed in a controlled laboratory environment. This setting incorporates many realistic aspects of live networks, including commercial IP routers, commodity workstations and a range of traffic conditions, and provides the critical ability to compare SLAM output with "ground truth". Performance tests with SLAM in the live Internet are also a subject of future work. Another key question is the following: given a daily (or based on some other time scale) budget of probes that may be used to monitor compliance with a SLA, what are the considerations for optimizing the probe process? Should the probing period be over a relatively long time scale (*e.g.*, the entire interval of interest), thus potentially limiting the accuracy of estimates, or should the probing period be over a shorter time scale, potentially improving estimation accuracy but at the cost of not probing over the entire interval, thus potentially missing important events? We intend to consider this issue in future work.

References

[1] A. Shaikh and A. Greenberg, "Operations and Management of IP Networks: What Researchers Should Know. Tutorial Session, ACM SIGCOMM '05. http://www.research.att.com/~albert/sigcomm05-greenberg-shaikh-tute.pdf, August, 2005."

[2] "Sprint NEXTEL service level agreements," http://www.sprint.com/business/support/serviceLevelAgreements.jsp, 2006.

[3] "AT&T Managed Internet Service (MIS)," http://new.serviceguide.att.com/mis.htm, 2006.

[4] "NTT Communications Global IP Network Service Level Agreement (SLA)," http://www.ntt.net/products/sla/sla_ts.cfm, 2006.

[5] M. Roughan, "Fundamental bounds on the accuracy of network performance measurements," in *ACM SIGMETRICS*, June 2005.

[6] P. Barford and J. Sommers, "Comparing probe- and router-based packet loss measurements," *IEEE Internet Computing*, September/October 2004.

[7] J. Sommers, P. Barford, N. Duffield, and A. Ron, "Improving accuracy in end-to-end packet loss measurement," in *Proceedings of ACM SIGCOMM '05*, 2005.

[8] J. Martin and A. Nilsson, "On service level agreements for IP networks," in *IEEE INFOCOM 2002*, vol. 2, 2002.

[9] L. Ciavattone, A. Morton, and G. Ramachandran, "Standardized active measurements on a tier 1 IP backbone," *IEEE Communications*, vol. 41, no. 6, pp. 90–97, June 2003.

[10] T. Zseby, "Deployment of sampling methods for SLA validation with non-intrusive measurements," in *Proceedings of Passive and Active Measurement Workshop*, 2001.

[11] M. Chan, Y. Lin, and X. Wang, "A scalable monitoring approach for service level agreements validation," in *IEEE International Conference on Network Protocols (ICNP)*, 2000, pp. 37–48.

[12] M. Aida, N. Miyoshi, and K. Ishibashi, "A scalable and lightweight QoS monitoring technique combining passive and active approaches," in *Proceedings of IEEE INFOCOM '03*, March 2003.

[13] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," in *Proceedings of ACM SIGCOMM '93*, San Francisco, September 1993.

[14] A. Pasztor and D. Veitch, "A precision infrastructure for active probing," in *Passive and Active Measurement Workshop*, 2001.

[15] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the constancy of Internet path properties," in *Proceedings of ACM SIGCOMM Internet Measurement Workshop '01*, San Francisco, November 2001.

[16] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, "Measurement and modeling of temporal dependence in packet loss," in *Proceedings of IEEE INFOCOM '99*, New York, NY, March 1999.

[17] V. Paxson, "Measurements and analysis of end-to-end Internet dynamics," Ph.D. dissertation, University of California Berkeley, 1997.

[18] Internet Protocol Performance Metrics, "http://www.advanced.org/ippm/index.html," 2006.

[19] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis, "Framework for IP performance metrics," IETF RFC 2330, 1998.

[20] G. Almes, S. Kalidindi, and M. Zekauskas, "A one way packet loss metric for IPPM," IETF RFC 2680, September 1999.

[21] J. Sommers, P. Barford, N. Duffield, and A. Ron, "A geometric approach to improving active packet loss measurement," December 2005, submitted to ACM/IEEE Transactions on Networking.

[22] S. Agarwal, J. Sommers, and P. Barford, "Scalable network path emulation," in *Proceedings of IEEE MASCOTS '05*, September 2005.

[23] E. Corell, P. Saxholm, and D. Veitch, "A user friendly TSC clock," in *Proceedings of Passive and Active Measurement Conference*, March 2006.

[24] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs, "Iperf 1.7.0 – the TCP/UDP bandwidth measurement tool," http://dast.nlanr.net/Projects/Iperf, 2006.

[25] J. Sommers and P. Barford, "Self-configuring network traffic generation," in *Proceedings of ACM SIGCOMM Internet Measurement Conference '04*, 2004.